

DB2 for z/OS  
Übungen zur Schulung

16. August 2013

Eine Ausarbeitung von:

**cps4it**

Ralf Seidler • Stromberger Straße 36A • 55411 Bingen  
Fon: +49-6721-992611 • Fax: -49-6721-992613 • Mail: [rseidler@cps4it.de](mailto:rseidler@cps4it.de)  
Internet : <http://www.cps4it.de>  
Steuernummer: 08/220/2497/3, Finanzamt Bingen, Ust-ID : DE214792185

# Inhaltsverzeichnis

<b>1</b>	<b>VORBEREITUNGEN.....</b>	<b>4</b>
1.1	ANMELDEN UND TEST DER USER-IDENT .....	4
1.2	BIBLIOTHEK FÜR ÜBUNGSAUFGABEN ERSTELLEN .....	4
1.3	EINSTIEG IN DAS DB2-MENÜ .....	4
1.4	ÜBUNGSDATENBANKEN .....	5
1.4.1	<i>Datenbank Lieferant</i> .....	5
1.4.2	<i>Datenbank Teil</i> .....	5
1.4.3	<i>Datenbank Projekt</i> .....	6
1.4.4	<i>Datenbank Auftrag</i> .....	6
1.5	BEISPIELE .....	7
1.5.1	<i>Lieferantennummer und Status</i> .....	7
1.5.2	<i>ein Feld auswählen</i> .....	7
1.5.3	<i>ein Feld auswählen – ohne Duplikate</i> .....	7
1.5.4	<i>Berechnung und feste Werte</i> .....	8
1.5.5	<i>alle Daten</i> .....	8
1.5.6	<i>Bedingung</i> .....	9
1.5.7	<i>Sortierung</i> .....	9
1.5.8	<i>Begrenzung (1)</i> .....	9
1.5.9	<i>Begrenzung (2)</i> .....	10
1.5.10	<i>Wertetabelle (1)</i> .....	10
1.5.11	<i>Wertetabelle (2)</i> .....	10
1.5.12	<i>Wertetabelle (3)</i> .....	11
1.5.13	<i>Wertetabelle (4)</i> .....	11
1.5.14	<i>NULL</i> .....	12
1.5.15	<i>JOIN / INNER JOIN</i> .....	13
1.5.16	<i>Kartesisches Produkt</i> .....	13
1.5.17	<i>FULL OUTER JOIN</i> .....	13
1.5.18	<i>LEFT OUTER JOIN</i> .....	14
1.5.19	<i>RIGHT OUTER JOIN</i> .....	14
1.5.20	<i>Subquery (1)</i> .....	14
1.5.21	<i>Subquery (2)</i> .....	15
1.5.22	<i>COUNT</i> .....	15
1.5.23	<i>SUM</i> .....	15
1.5.24	<i>Gruppieren</i> .....	16
1.5.25	<i>Gruppen mit Eigenschaften</i> .....	16
1.5.26	<i>Kombination von Abfragen</i> .....	16
1.5.27	<i>Einfügen von einzelnen Zeilen</i> .....	17
1.5.28	<i>Einfügen von mehreren Zeilen aus SELECT</i> .....	17
1.5.29	<i>Update von Zeilen</i> .....	17
1.5.30	<i>Löschen von Zeilen</i> .....	18
1.5.31	<i>Abfrage auf Systemtabelle (1)</i> .....	18
1.5.32	<i>Abfrage auf Systemtabelle (2)</i> .....	18
<b>2</b>	<b>DATENBANKEN ERSTELLEN .....</b>	<b>19</b>
2.1	TABELLE ERSTELLEN .....	19
2.2	INDEX ERSTELLEN .....	19
2.3	SYNONYM.....	19
<b>3</b>	<b>DATEN IN TABELLEN FINDEN.....</b>	<b>20</b>
3.1	PROJEKTDATEN ALLER PROJEKTE.....	20
3.2	PROJEKTDATEN EINER LOKATION.....	20
3.3	SORTIEREN .....	20
3.4	AUSWAHL.....	20

<b>4</b>	<b>JOIN .....</b>	<b>21</b>
4.1	AUFTRÄGE MIT NAMEN DER LIEFERANTEN .....	21
4.2	TEILE VON BESTIMMTEN LIEFERANTEN .....	21
4.3	ZUSÄTZLICHE ZEILEN .....	21
4.4	JOIN STATT SUBQUERY .....	21
<b>5</b>	<b>FUNKTIONEN .....</b>	<b>22</b>
5.1	STATUS VON LIEFERANTEN .....	22
5.2	ANZAHL VON LIEFERANTEN FÜR PROJEKTE .....	22
5.3	GRUPPIERUNG VON TEILEN .....	22
5.4	ORTE SUCHEN.....	22
<b>6</b>	<b>ÄNDERN VON TABELLEN (1).....</b>	<b>23</b>
6.1	EIGENE TABELLENINHALTE ERZEUGEN .....	23
6.2	ÄNDERN VON ZEILEN .....	23
6.3	LÖSCHEN VON ZEILEN.....	23
6.4	LÖSCHEN VON TABELLENINHALTEN.....	23
<b>7</b>	<b>ÄNDERN VON TABELLEN (2).....</b>	<b>24</b>
7.1	EIGENE TABELLENINHALTE ERZEUGEN .....	24
7.2	ÄNDERN VON SPALTEN .....	24
7.3	LÖSCHEN VON ZEILEN.....	24
7.4	EINFÜGEN VON ZEILEN.....	24
7.5	ERSTELLEN EINER VIEW .....	25
<b>8</b>	<b>SYSTEMTABELLEN.....</b>	<b>25</b>
8.1	SYSTEMTABELLEN AUSLESEN.....	25
<b>9</b>	<b>DB2 UTILITIES .....</b>	<b>26</b>
9.1	TABELLE VOLLSTÄNDIG ENTLADEN.....	26
9.2	TABELLE TEILWEISE ENTLADEN .....	26
9.3	RUNSTATS ERZEUGEN.....	26
9.4	BMC MASTERMIND / DB2 CATALOG MANAGER .....	26
9.5	QUICKSTART.....	26
9.6	FILEAID DB2 .....	26
<b>10</b>	<b>SQL IM ANWENDUNGSPROGRAMM.....</b>	<b>27</b>
10.1	PROGRAMM MIT DB2-BEFEHLEN SCHREIBEN .....	27
10.2	TABELLE LESEN .....	27
<b>11</b>	<b>CURSOR-VERARBEITUNG .....</b>	<b>28</b>
11.1	TABELLE LESEN .....	28
11.2	TABELLE LESEN UND 1 ZEILE VERÄNDERN .....	28
11.3	TABELLE LESEN MIT MULTI-ROW-FETCH .....	28

## **1 Vorbereitungen**

---

### **1.1 Anmelden und Test der User-Iden**

---

Melden Sie sich nach Vorgabe im TSO an.  
Melden Sie sich wieder ab und erneut wieder an.  
Konfigurieren Sie Ihre TSO-Session, wie Sie am besten arbeiten können.

### **1.2 Bibliothek für Übungsaufgaben erstellen**

---

Erstellen Sie eine PO-Datei / Library, um die Übungsaufgaben dieses Seminars aufnehmen zu können.

Name der Bibliothek	userid.KURS.SPUFIIN
Satzlänge	80 Byte
Satzformat	fest geblockt
Dateigröße	5 Spuren primär, 2 Spuren sekundär
Directory	Platz für 30 Member

Kontrollieren Sie das Ergebnis.

### **1.3 Einstieg in das DB2-Menü**

---

Testen Sie den Zugang zum DB2-Menü

---

## 1.4 Übungsdatenbanken

---

### 1.4.1 Tabelle Lieferant

---

LNR	LNAME	LSTATUS	ORT
L1	NEUMANN	30	BERLIN
L2	SCHMIDT	20	HAMBURG
L3	KRAUSE	30	HAMBURG
L4	MEIER	10	BERLIN
L5	SCHULZ	20	FRANKFURT

LNR ist Primär-Schlüssel

### 1.4.2 Tabelle Teil

---

TNR	TNAME	FARBE	GEWICHT	ORT
T1	C	BLAU	19	BERLIN
T2	D	GELB	12	HAMBURG
T3	S	ROT	14	STUTTGART
T4	S	BLAU	17	BERLIN
T5	B	ROT	17	HAMBURG
T6	N	BLAU	12	BERLIN

TNR ist Primär-Schlüssel

### 1.4.3 Tabelle Projekt

PNR	PNAME	ORT
P1	S4711	HAMBURG
P2	P0815	STUTTGART
P3	R111	FRANKFURT
P4	C007	FRANKFURT
P5	C1234	BERLIN
P6	TZ1	MAINZ
P7	TX25	BERLIN
P8	P888	MUENCHEN

PNR ist Primär-Schlüssel

### 1.4.4 Tabelle Auftrag

LNR	TNR	PNR	MENGE
L1	T1	P1	300
L1	T1	P4	800
L2	T3	P1	500
L2	T3	P2	300
L2	T3	P3	300
L2	T3	P4	600
L2	T3	P5	700
L2	T3	P6	500
L2	T3	P7	900
L2	T5	P2	200
L3	T3	P1	300
L3	T4	P2	600
L4	T6	P3	400
L4	T6	P7	400
L5	T2	P2	300
L5	T2	P4	100
L5	T5	P5	500
L5	T5	P7	100
L5	T6	P2	200
L5	T1	P4	100
L5	T3	P4	200
L5	T4	P4	800
L5	T5	P4	400
L5	T6	P4	500

Die Werte LNR, TNR und PNR bilden zusammen die eindeutige Identifikation eines Auftrags. Sie stellen die Auftragsnummer dar. Das heißt, dass die Spalten LNR, TNR und PNR zusammen den Primär-Schlüssel bilden.

## 1.5 Beispiele

---

Hinweis: Sollte es nicht möglich sein, mit den Synonymen L, T, LTP, A zu arbeiten, dann ist mit LIEFERANT, TEIL, AUFTRAG oder, wenn dieses ebenfalls nicht gehen sollte, mit ref-u-id.LIEFERANT, ref-u-id.TEIL, ref-u-id.AUFTRAG zu arbeiten.

### 1.5.1 Lieferantenummer und Status

---

- Aufgabe  
auswählen von Lieferanten-Nummer und Status der Lieferanten aus Berlin

- Befehl (BSP01)  

```
SELECT LNR, LSTATUS
      FROM L
     WHERE ORT = 'BERLIN'
```

- Ergebnis:  

```
LNR LSTATUS
--- -
L1      30
L4      10
```

### 1.5.2 ein Feld auswählen

---

- Aufgabe  
auswählen aller Teilenummern der bestellten Materialien

- Befehl (BSP02)  

```
SELECT TNR
      FROM LT
```

- Ergebnis:  

```
TNR
---
T1      T6      T4
T2      T1      T5
T3      T2
T4      T2
T5      T2
...      ...
```

### 1.5.3 ein Feld auswählen – ohne Duplikate

---

- Aufgabe  
auswählen aller Teilenummern der bestellten Materialien ohne Duplikate

- Befehl (BSP03)  

```
SELECT DISTINCT TNR
      FROM LT
```

- Ergebnis:  

```
TNR
---
T1
T2
T3
T4
```

T5  
T6

#### 1.5.4 Berechnung und feste Werte

---

- **Aufgabe**  
Für alle Materialien ist die Teilenummer und das aus dem Nettogewicht und einem konstanten Gewichtungsfaktor errechnete Bruttogewicht aufzulisten. Außerdem soll die Formel angezeigt werden.

- **Befehl (BSP04)**

```
SELECT TNR, 'Nettogewicht x 1,25 = ',  
        GEWICHT * 1.25  
FROM T
```

- **Ergebnis:**

```
TNR  
-----  
T1  Nettogewicht x 1,25 = 23.75  
T2  Nettogewicht x 1,25 = 15.00  
T3  Nettogewicht x 1,25 = 17.50  
T4  Nettogewicht x 1,25 = 21.25  
T5  Nettogewicht x 1,25 = 21.25  
T6  Nettogewicht x 1,25 = 15.00
```

#### 1.5.5 alle Daten

---

- **Aufgabe**  
Auflisten aller Daten der Lieferanten-Tabelle

- **Befehl (BSP05)**

```
SELECT *  
FROM L
```

oder:

```
SELECT LNR, LNAME, LSTATUS, ORT  
FROM L
```

- **Ergebnis:**

```
LNR  LNAME      LSTATUS  ORT  
-----  
L1   NEUMANN    30      BERLIN  
L2   SCHMIDT    20      HAMBURG  
L3   KRAUSE     30      HAMBURG  
L4   MEIER      10      BERLIN  
L5   SCHULZ     20      FRANKFURT
```



### 1.5.6 Bedingung

- Aufgabe  
Auflisten der LNR der Lieferanten in Hamburg mit einem Status größer als 20.

- Befehl (BSP06)

```
SELECT LNR
      FROM L
     WHERE ORT      = 'Hamburg'
           AND LSTATUS > 20
```

- Ergebnis:

```
LNR
---
L3
```

### 1.5.7 Sortierung

- Aufgabe  
Aufsuchen der Lieferantenummer der Lieferanten in Hamburg und Anzeige absteigend sortiert nach dem Lieferantenstatus.

- Befehl (BSP07)

```
SELECT      LNR, LSTATUS
      FROM    L
     WHERE    ORT      = 'Hamburg'
           ORDER BY LSTATUS DESC
```

- Ergebnis:

```
LNR LSTATUS
--- -
L3      30
L2      20
```

### 1.5.8 Begrenzung (1)

- Aufgabe  
Liste die Teile auf, deren Gewicht zwischen 16 und 19 kg liegt.

- Befehl (BSP08)

```
SELECT      *
      FROM    T
     WHERE    GEWICHT BETWEEN 16 AND 19
```

- Ergebnis:

```
TNR TNAME      FARBE  GEWICHT  ORT
--- -
T1  C          BLAU    19     BERLIN
T4  S          BLAU    17     BERLIN
T5  B          ROT     17     HAMBURG
```

---

### 1.5.9 Begrenzung (2)

---

- Aufgabe  
Liste die Teile auf, deren Gewicht nicht zwischen 16 und 19 kg liegt.

- Befehl (BSP09)

```
SELECT *
FROM T
WHERE GEWICHT NOT BETWEEN 16 AND 19
```

- Ergebnis:

TNR	TNAME	FARBE	GEWICHT	ORT
T2	D	GELB	12	HAMBURG
T3	S	ROT	14	STUTTGART
T6	N	BLAU	12	BERLIN

---

### 1.5.10 Wertetabelle (1)

---

- Aufgabe  
Liste der Teile mit dem Gewicht 12, 13 oder 17 kg.

- Befehl (BSP10)

```
SELECT *
FROM T
WHERE GEWICHT IN (12, 13, 17)
```

- Ergebnis:

TNR	TNAME	FARBE	GEWICHT	ORT
T2	D	GELB	12	HAMBURG
T4	S	BLAU	17	BERLIN
T5	B	ROT	17	HAMBURG
T6	N	BLAU	12	BERLIN

---

### 1.5.11 Wertetabelle (2)

---

- Aufgabe  
Liste der Teile mit dem Gewicht 12, 13 oder 17 kg.

- Befehl (BSP11)

```
SELECT *
FROM T
WHERE GEWICHT = 12
OR GEWICHT = 13
OR GEWICHT = 17
```

- Ergebnis:

TNR	TNAME	FARBE	GEWICHT	ORT
T2	D	GELB	12	HAMBURG
T4	S	BLAU	17	BERLIN
T5	B	ROT	17	HAMBURG
T6	N	BLAU	12	BERLIN

---

**1.5.12 Wertetabelle (3)**

---

- Aufgabe  
Liste der Teile, die nicht das Gewicht 12, 13 oder 17 kg haben.

- Befehl (BSP12)

```
SELECT *
FROM T
WHERE GEWICHT NOT IN (12, 13, 17)
```

- Ergebnis:

TNR	TNAME	FARBE	GEWICHT	ORT
T1	C	BLAU	19	BERLIN
T3	S	ROT	14	STUTTGART

---

**1.5.13 Wertetabelle (4)**

---

- Aufgabe  
Liste der Teile, die nicht das Gewicht 12, 13 oder 17 kg haben.

- Befehl (BSP13)

```
SELECT *
FROM T
WHERE GEWICHT ^= 12
AND GEWICHT ^= 13
AND GEWICHT ^= 17
```

- Ergebnis:

TNR	TNAME	FARBE	GEWICHT	ORT
T1	C	BLAU	19	BERLIN
T3	S	ROT	14	STUTTGART

### 1.5.14 NULL

---

Annahme: Lieferant L3 hat NULL statt '30'

- Aufgabe  
Liste der Lieferantennummern der Lieferanten mit einem Status > '25'.

- Befehl (BSP14)

```
SELECT      LNR
FROM        L
WHERE       LSTATUS > 25
```

- Ergebnis:

```
LNR
---
L1
```

Annahme: Lieferant L3 hat NULL statt '30'

- Aufgabe  
Liste der Lieferantennummern der Lieferanten mit einem Status <= '25'.

- Befehl (BSP14)

```
SELECT      LNR
FROM        L
WHERE       LSTATUS <= 25
```

- Ergebnis:

```
LNR
---
L2
L4
L5
```

Annahme: Lieferant L3 hat NULL statt '30'

- Aufgabe  
Liste der Lieferantennummern der Lieferanten mit einem Status NULL.

- Befehl (BSP14)

```
SELECT      LNR
FROM        L
WHERE       LSTATUS IS NULL
```

- Ergebnis:

```
LNR
---
L3
```

### 1.5.15 JOIN / INNER JOIN

- Aufgabe  
Liste der Lieferantennummern der Lieferanten mit einem Status NULL.
- Befehl (BSP15)  

```
SELECT      *
      FROM    L, T
      WHERE   L.ORT = T.ORT
```

 oder:  

```
SELECT      *
      FROM    L INNER JOIN T
      ON      L.ORT = T.ORT
```
- Ergebnis:  

```
LNR
---
L3
```

### 1.5.16 Kartesisches Produkt

- Aufgabe  
Anzeige aller Daten aus den Tabellen Lieferant und Teil. Was ist wirklich gewollt?
- Befehl (BSP16)  

```
SELECT      *
      FROM    L, T
      WHERE   L.ORT = T.ORT
```
- Ergebnis:  
Egal ;-)

### 1.5.17 FULL OUTER JOIN

- Aufgabe  
Kombiniere Daten aus den Tabellen L und T, bei denen der Ort des Lieferanten gleich dem Ort des Teilelagers ist. Zusätzlich sollen die Zeilen ausgegeben werden, die keine Übereinstimmungen haben.
- Befehl (BSP17)  

```
SELECT      *
      FROM    L FULL OUTER JOIN T
      ON      L.ORT = T.ORT
```
- Ergebnis:

LNR	LNAME	LSTATUS	ORT	TNR	TNAME	FARBE	GEWICHT	ORT
L1	NEUMANN	30	BERLIN	T1	C	BLAU	19	BERLIN
L1	NEUMANN	30	BERLIN	T4	S	BLAU	17	BERLIN
L1	NEUMANN	30	BERLIN	T6	N	BLAU	12	BERLIN
L2	SCHMIDT	20	HAMBURG	T2	D	GELB	12	HAMBURG
L2	SCHMIDT	20	HAMBURG	T5	B	ROT	17	HAMBURG
L3	KRAUSE	30	HAMBURG	T2	D	GELB	12	HAMBURG
L3	KRAUSE	30	HAMBURG	T5	B	ROT	17	HAMBURG
L4	MEIER	10	BERLIN	T1	C	BLAU	19	BERLIN
L4	MEIER	10	BERLIN	T4	S	BLAU	17	BERLIN
L4	MEIER	10	BERLIN	T6	N	BLAU	12	BERLIN
L5	SCHULZ	20	FRANKFURT					
				T3	S	ROT	14	STUTTGART

---

**1.5.18 LEFT OUTER JOIN**

---

- Aufgabe  
Kombiniere Daten aus den Tabellen L und T, bei denen der Ort des Lieferanten gleich dem Ort des Teilelagers ist. Zusätzlich sollen die Zeilen aus L ausgegeben werden, die keine Übereinstimmung in T haben.
- Befehl (BSP18)

```
SELECT      *
  FROM      L LEFT OUTER JOIN T
  ON        L.ORT = T.ORT
```
- Ergebnis:  
. . .

---

**1.5.19 RIGHT OUTER JOIN**

---

- Aufgabe  
Kombiniere Daten aus den Tabellen L und T, bei denen der Ort des Lieferanten gleich dem Ort des Teilelagers ist. Zusätzlich sollen die Zeilen aus L ausgegeben werden, die keine Übereinstimmung in T haben.
- Befehl (BSP19)

```
SELECT      *
  FROM      L RIGHT OUTER JOIN T
  ON        L.ORT = T.ORT
```
- Ergebnis:  
. . .

---

**1.5.20 Subquery (1)**

---

- Aufgabe  
Liste der Namen aller Lieferanten, die Teil T2 liefern.
- Befehl (BSP20)

```
SELECT      LNAME
  FROM      L
  WHERE     LNR IN (SELECT LNR
                   FROM LT
                   WHERE TNR = 'T3')
```

```
SELECT      L.LNAME
  FROM      L INNER JOIN LT
  ON        L.LNR = LT.LNR
  WHERE     LT.TNR = 'T2'
```
- Ergebnis:  
. . .

### 1.5.21 Subquery (2)

---

- Aufgabe  
Benötigt wird eine Liste der Lieferanten, welchen mindestens ein Teil mit der Farbe ROT liefern.

- Befehl (BSP21)

```
SELECT      LNAME
  FROM      L
 WHERE     LNR IN
           ( SELECT LNR
             FROM LT
             WHERE TNR IN
               ( SELECT TNR
                 FROM T
                 WHERE FARBE = 'ROT' )
           )
```

- Ergebnis:

. . .

### 1.5.22 COUNT

---

- Aufgabe  
Einfache Beispiele für COUNT

- Befehl (BSP22)

```
SELECT COUNT (*)
  FROM L
```

```
SELECT COUNT (*)
  FROM LT
 WHERE TNR = 'T2'
```

```
SELECT      COUNT (LNR)
  FROM      LT
```

```
SELECT      COUNT (DISTINCT LNR)
  FROM      LT
```

- Ergebnis:

. . .

### 1.5.23 SUM

---

- Aufgabe  
Einfaches Beispiele für SUM

- Befehl (BSP23)

```
SELECT      SUM (MENGE)
  FROM      LT
 WHERE     TNR = 'T4'
```

- Ergebnis:

. . .

### 1.5.24 Gruppieren

---

- Aufgabe  
Zeige je Teil die Summe der im Auftrag befindlichen Menge an.
- Befehl (BSP24)

```
SELECT      TNR, SUM(MENGE)
  FROM      LT
  GROUP BY  TNR
```
- Ergebnis:  
. . .

### 1.5.25 Gruppen mit Eigenschaften

---

- Aufgabe  
Zeige je Teil die Summe der im Auftrag befindlichen Menge an.
- Befehl (BSP25)

```
SELECT      TNR, SUM(MENGE)
  FROM      LT
  GROUP BY  TNR
```
- Ergebnis:  
. . .

### 1.5.26 Kombination von Abfragen

---

- Aufgabe  
Liste die Teilenummern der Teile auf, die entweder ein Gewicht über 16 kg haben oder vom Lieferanten L1 geliefert werden.
- Befehl (BSP26)

```
SELECT      TNR
  FROM      T
  WHERE     GEWICHT > 16
UNION
SELECT      TNR
  FROM      LT
  WHERE     LNR = 'L1'

SELECT      TNR
  FROM      T
  WHERE     GEWICHT > 16
UNION ALL
SELECT      TNR
  FROM      LT
  WHERE     LNR = 'L1'
```
- Ergebnis:  
. . .



---

### 1.5.27 Einfügen von einzelnen Zeilen

---

- Aufgabe  
einfaches Beispiel
- Befehl (BSP27)  

```
INSERT
  INTO T (TNR, ORT, GEWICHT)
VALUES ('T7', 'DORTMUND', 21)
```
- Ergebnis:  
. . .

---

### 1.5.28 Einfügen von mehreren Zeilen aus SELECT

---

- Aufgabe  
einfaches Beispiel
- Befehl (BSP28)  

```
CREATE TABLE TEMP (TNR CHAR(5)
                   , GESMENGE INTEGER)
--
INSERT
  INTO TEMP (TNR, GESMENGE)
  SELECT TNR, SUM(MENGE)
  FROM LT
  GROUP BY TNR
--
DROP TABLE TEMP
```
- Ergebnis:  
. . .

---

### 1.5.29 Update von Zeilen

---

- Aufgabe  
Ändern der Zeile mit der Teilenummer T6 in der Spalte FARBE nach BLAU, das Gewicht soll um 2 kg erhöht werden und der Ort soll NULL sein.
- Befehl (BSP29)  

```
UPDATE T
  SET FARBE = 'BLAU',
      GEWICHT = GEWICHT + 2,
      ORT = NULL
WHERE TNR = 'T7'
```
- Ergebnis:  
. . .
- Aufgabe  
Verdopple den Status aller Lieferanten in BERLIN
- Befehl (BSP29)  

```
UPDATE L
  SET LSTATUS = 2 * LSTATUS
WHERE ORT = 'BERLIN'
```
- Ergebnis:  
. . .

### 1.5.30 Löschen von Zeilen

---

- Aufgabe  
Löschen der Teile mit TNR = T7
- Befehl (BSP30)  
DELETE  
FROM T  
WHERE TNR = 'T7'

### 1.5.31 Abfrage auf Systemtabelle (1)

---

- Aufgabe  
Welche Tabellen enthalten Spalten mit dem Namen LNR?
- Befehl (BSP31)  
SELECT TBNAME  
FROM SYSIBM.SYSCOLUMNS  
WHERE NAME = 'LNR'

### 1.5.32 Abfrage auf Systemtabelle (2)

---

- Aufgabe  
Welche Spalten hat die Tabelle LIEFERANT?
- Befehl (BSP32)  
SELECT NAME  
FROM SYSIBM.SYSCOLUMNS  
WHERE TBNAME = 'LIEFERANT'

## **2 Datenbanken erstellen**

---

### **2.1 Tabelle erstellen**

---

Member: SPIN201

Schreiben Sie den SQL, der die Tabelle PROJEKT definiert. Die Projekte werden durch die Projektnummer eindeutig identifiziert. Das zugehörige Feld heißt PNR und ist 5 Stellen alphanumerisch. Von einer anderen Tabelle aus soll auf Ihre Tabelle verwiesen werden. Die von der Projektnummer abhängigen Attribute sind der Name des Projektes mit dem Feldnamen PNAME sowie der Ort mit dem Feldnamen ORT. PNAME ist 8 Stellen lang und alphanumerisch, Ort ist 10 Stellen lang und ebenfalls alphanumerisch.

### **2.2 Index erstellen**

---

Member: SPIN202

Die Tabelle Projekt soll nur eindeutig zu identifizierende Zeilen enthalten. Definieren Sie den notwendigen Index, um die Eindeutigkeit bei Hinzufügen und Ändern zu erzwingen. Der verkürzte Namen des Index soll PBRX sein.

### **2.3 Synonym**

---

Member: SPIN203

Bilden Sie für die Tabellen LIEFERANT, TEIL, PROJEKT und AUFTRAG Synonyme mit den Namen L, T, P, LTP oder A.

### **3 Daten in Tabellen finden**

---

#### **3.1 Projektdaten aller Projekte**

---

Member: SPIN301

Erstellen Sie eine Projektliste mit allen Daten der Tabelle PROJEKT. Lösen Sie die Aufgabe auf 2 unterschiedlichen Wegen. Welcher Weg ist für einen Test sinnvoller? Welcher Weg ist für die Aufnahme in ein Programm sinnvoller?

#### **3.2 Projektdaten einer Lokation**

---

Member: SPIN302

Listen Sie alle Daten der Projekte in Berlin auf. Lösen Sie die Aufgabe auf 2 unterschiedlichen Wegen. Welcher Weg ist für einen Test sinnvoller? Welcher Weg ist für die Aufnahme in ein Programm sinnvoller?

#### **3.3 Sortieren**

---

Member: SPIN303

Suchen Sie die Lieferantennummern aller Lieferanten in sortierter Reihenfolge, die für die Projekte P1, P4 oder P6 Teile liefern. Lösen Sie diese Aufgabe auf 2 unterschiedlichen Wegen. Welche ist vorzuziehen?

#### **3.4 Auswahl**

---

Member: SPIN304

Geben Sie die Aufträge aus, deren gelieferte Mengen zwischen 400 und 800 liegen. Die Grenzen sind jeweils eingeschlossen.

## **4 JOIN**

---

### **4.1 Aufträge mit Namen der Lieferanten**

---

Member: SPIN401

Erstellen Sie eine Liste aller Aufträge mit den Namen der Lieferanten. Lösen Sie die Aufgabe auf 2 unterschiedlichen Wegen.

### **4.2 Teile von bestimmten Lieferanten**

---

Member: SPIN402

Listen Sie alle Teilenummern der Teile, die von Lieferanten aus Berlin geliefert werden. Lösen Sie die Aufgabe auf 2 unterschiedlichen Wegen.

### **4.3 zusätzliche Zeilen**

---

Member: SPIN403

Listen Sie alle Aufträge mit ihren Projektdaten auf. Außerdem sollen in der Liste die Projektdaten angezeigt werden, für die es keine Aufträge gibt.

### **4.4 Join statt Subquery**

---

Member: SPIN404

Benötigt wird eine Liste der Lieferanten, welchen mindestens ein Teil mit der Farbe ROT liefern. Lösen Sie diese Aufgabe mit einem Subquery aber auch mit einem Join.

## **5 Funktionen**

---

### **5.1 Status von Lieferanten**

---

Member: SPIN501

Erstellen Sie eine Liste der Lieferantennummern von denjenigen Lieferanten, die einen niedrigeren Status als der Lieferant L1 haben.

### **5.2 Anzahl von Lieferanten für Projekte**

---

Member: SPIN502

Wie viele Projekte werden von dem Lieferanten L1 beliefert?

### **5.3 Gruppierung von Teilen**

---

Member: SPIN503

Erstellen Sie eine Liste mit den Teilen, die für mindestens 1 Projekt geliefert werden. Zu jedem Teil geben Sie die Teilenummer, die Projektnummer und die jeweilige Gesamtmenge an.

### **5.4 Orte suchen**

---

Member: SPIN504

Listen Sie die Namen der Orte in sortierter Reihenfolge auf, in denen wenigstens ein Lieferant, ein Teil oder ein Projekt seinen Ort hat.

## **6 Ändern von Tabellen (1)**

---

### **6.1 Eigene Tabelleninhalte erzeugen**

---

Member: SPIN601

Erstellen Sie in den vorliegenden Tabellen L, T, LT, P jeweils eigene Zeilen. Sofern Sie mit den Tabellen des Referenten arbeiten, wählen Sie dazu Schlüssel, die nur Ihnen zugeordnet worden sind.

### **6.2 Ändern von Zeilen**

---

Member: SPIN602

Spielen Sie mit Ihren Tabelleninhalten, indem Sie die Inhalte der Spalten verändern. Prüfen Sie Ihre Änderungen, ob sie Ihren Erwartungen entsprechen.

### **6.3 Löschen von Zeilen**

---

Member: SPIN603

Spielen Sie mit Ihren Tabelleninhalten, indem Sie einzelne Tabellenzeilen löschen. Prüfen Sie Ihre Änderungen, ob sie Ihren Erwartungen entsprechen.

### **6.4 Löschen von Tabelleninhalten**

---

Member: SPIN604

Löschen Sie die von Ihnen erzeugten Tabelleninhalte wieder.

## **7 Ändern von Tabellen (2)**

---

### **7.1 Eigene Tabelleninhalte erzeugen**

---

Member: SPIN701

Erstellen Sie in den vorliegenden Tabellen L, T, LT, P jeweils eigene Zeilen. Sofern Sie mit den Tabellen des Referenten arbeiten, wählen Sie dazu Schlüssel, die nur Ihnen zugeordnet worden sind. Die Zeilen sollen bis auf die Schlüssel echte Duplikate der ursprünglich vorhandenen sein.

### **7.2 Ändern von Spalten**

---

Member: SPIN702

Setzen Sie den Status aller Lieferanten in Berlin auf 40. Prüfen Sie das Ergebnis.

### **7.3 Löschen von Zeilen**

---

Member: SPIN703

Löschen Sie alle Lieferanten in Hamburg, deren Status kleiner als 25 ist. Prüfen Sie das Ergebnis.

### **7.4 Einfügen von Zeilen**

---

Member: SPIN704

Der neue Lieferant Schulze in Münster ist mit der Lieferantenummer L10 in die Lieferantentabelle einzutragen. Sein Lieferantenstatus ist nicht bekannt. Lösen Sie die Aufgabe auf 2 unterschiedlichen Wegen.



## 7.5 Erstellen einer View

---

Member: SPIN705

Erstellen Sie die View VAUFTRAG mit den folgenden Spalten aus den Tabellen LIEFERANT, TEIL und AUFTRAG:

LNR, LNAME, TNR, TNAME, MENGE

Die Spaltennamen in der View sollen für LNAME LIEF\_NAME und für TNAME TEIL\_NAME heißen.

## 8 Systemtabellen

---

### 8.1 Systemtabellen auslesen

---

Member: SPIN801

Lesen Sie verschiedene Informationen zu den Beispieltabellen aus den Systemtabellen aus. Prüfen Sie, ob die Ergebnisse Ihren Erwartungen entsprechen. Behelfen Sie sich bzgl. der Spaltennamen der Systemtabellen der Systemtabellen selbst und beantworten Sie auch die Frage(n): Welche Columns haben die (manche) Systemtabellen?

## **9 DB2 Utilities**

---

### **9.1 Tabelle vollständig entladen**

---

Member: SPIN901

Entladen Sie die Tabelle LIEFERANT in eine sequentielle Datei.

### **9.2 Tabelle teilweise entladen**

---

Member: SPIN902

Entladen Sie Teile der Tabelle LIEFERANT in eine sequentielle Datei.

### **9.3 Runstats erzeugen**

---

Wie ist es möglich, die Statistik-Informationen zu der Tabelle LIEFERANT zu aktualisieren?

### **9.4 BMC Mastermind / DB2 Catalog Manager**

---

Machen Sie sich ein wenig mit dem DB2 Catalog Manager von BMC vertraut.

### **9.5 QuickStart**

---

Schauen Sie sich die Menüs von QuickStart an.

### **9.6 FileAid DB2**

---

„Spielen“ Sie ein wenig mit dem Tool FileAid for DB2.

## **10 SQL im Anwendungsprogramm**

---

### **10.1 Programm mit DB2-Befehlen schreiben**

---

Member: SPINA01 / P96N7xV

Schreiben Sie ein Programm, das fehlerfrei umgewandelt wird und nur SQLCA als DB2-Source enthält.

### **10.2 Tabelle lesen**

---

Member: SPINA02 / P96N7xV

Schreiben Sie ein Programm, das eine Zeile der Tabelle, die der Referent genannt hat, ausliest und anzeigt.

## 11 Cursor-Verarbeitung

---

### 11.1 Tabelle lesen

---

Member: SPINB01/ P96N7xW

Schreiben Sie ein Programm, das 50 Zeilen der Tabelle, die der Referent genannt hat, ausliest und anzeigt.

### 11.2 Tabelle lesen und 1 Zeile verändern

---

Member: SPINB02 / P96N7xX

Schreiben Sie ein Programm, das alle Zeilen der Tabelle, die der Referent genannt hat, ausliest und anzeigt. Mittendrin soll, wenn eine bestimmte Bedingung (frei wählbar) erfüllt ist, ein UPDATE der aktuellen Zeile erfolgen.

### 11.3 Tabelle lesen mit multi-row-fetch

---

Member: SPINB03 / P96N7xY

Schauen Sie in der SQL-Referenz, wie ein multi-row-fetch zu kodieren ist. Schreiben Sie ein Programm, das jeweils 10 Zeilen der Tabelle, die der Referent genannt hat, ausliest und anzeigt.